CLAIMS

1. A method for configuring a bus system having a plurality of bus segments with bus master devices and slave devices connected thereto, the bus segments connected by bus bridges, each bus bridge having a bridge ID, a plurality of internal registers and an address bitmap for controlling information flow through the bridge, the method comprising:

   (a) walking the bus system to discover the bus topology and the bus bridges that form that topology;

   (b) assigning a bridge ID to each discovered bridge; and

   (c) entering information into internal registers and address bitmap of each discovered bridge to control the flow of information between bus segments.

2. The method of claim 1 wherein the bus topology is a tree configuration and step (a) comprises performing a recursive procedure that configures each branch of the tree.

3. The method of claim 1 wherein the bus system has an address space and wherein step (a) comprises:

   (a1) probing the address space for slave devices.

4. The method of claim 3 wherein step (a1) comprises:

   (a1a) checking for a duplicate slave address when a slave device is located.

5. The method of claim 4 wherein step (a1) comprises:

   (a1b) inserting a slave address of a located slave device into a global address bitmap if the slave address is not a duplicate; and

(a1c) inserting the slave address into a tunnel list if the slave address is a duplicate.

6. The method of claim 5 wherein step (a1) further comprises

(a1d) repeatedly probing the address space for upstream bridges when no slave device is located.

7. The method of claim 6 wherein step (b) comprises assigning a bridge ID to each located upstream bridge.

8. The method of claim 7 wherein step (a1) further comprises

(a1e) repeatedly probing for downstream bridges when no further upstream bridges are located in step (a1d).

9. The method of claim 8 wherein step (b) comprises assigning a bridge ID to each located downstream bridge.

10. The method of claim 9 wherein step (c) comprises entering information into internal registers and address bitmap of at least one downstream bridge when no further downstream bridges are detected in step (a1e).

11. The method of claim 1 further comprising:

(d) walking the bus system to discover upstream bridges; and

(e) entering information into internal registers and address bitmap of each discovered upstream bridge to control the flow of information between bus segments.

12. The method of claim 1 wherein step (b) comprises setting the bridge ID of all bridges to a predetermined bridge ID upon reset and successively configuring each bridge by sending commands and data to the predetermined bridge ID.

13. The method of claim 1 wherein step (b) comprises connecting all bridges on the same hierarchical level so that only one bridge at a time responds to the predetermined bridge ID.

14. The method of claim 13 wherein all bridges on the same hierarchical level are connected in a daisy chain configuration.

15. The method of claim 14 wherein a bridge in the daisy chain configuration enables the next bridge in the daisy chain configuration to respond to the predetermined bridge ID when the bridge is assigned a bridge ID other than the predetermined bridge ID.

16. The method of claim 1 wherein at least some of the bridges are bi-directional bridges comprised of two unidirectional bridges connected in parallel and wherein step (b) comprises giving the two unidirectional bridges different bridge IDs.

17. The method of claim 1 further comprising:

(f) providing additional information to each bridge to enable the bridge to operate with a deterministic arbitration protocol.

18. Apparatus for configuring a bus system having a plurality of bus segments with bus master devices and slave devices connected thereto, the bus segments connected by bus bridges, each bus bridge having a bridge ID, a plurality of

internal registers and an address bitmap for controlling information flow through the bridge, the apparatus comprising:

a configuration host that walks the bus system to discover the bus topology and the bus bridges that form that topology;

a mechanism that assigns a bridge ID to each discovered bridge; and

a mechanism that enters information into internal registers and address bitmap of each discovered bridge to control the flow of information between bus segments.

19. The apparatus of claim 18 wherein the bus topology is a tree configuration and the configuration host performs a recursive procedure that configures each branch of the tree.

20. The apparatus of claim 18 wherein the bus system has an address space and wherein the configuration host comprises a mechanism that probes the address space for slave devices.

21. The apparatus of claim 20 wherein the configuration host comprises a global address bitmap and a mechanism that uses the global address bitmap to check for a duplicate slave address when a slave device is located.

22. The apparatus of claim 21 wherein the configuration host comprises a mechanism that inserts a slave address of a located slave device into the global address bitmap if the slave address is not a duplicate; and inserts the slave address into a tunnel list if the slave address is a duplicate.

23. The apparatus of claim 22 wherein the configuration host comprises a mechanism that repeatedly probes the address space for upstream bridges when no slave device is located.

24. The apparatus of claim 23 wherein the bridge ID assigning mechanism comprises a mechanism that assigns a bridge ID to each located upstream bridge.

25. The apparatus of claim 24 wherein the configuration host further comprises a mechanism that repeatedly probes for downstream bridges when no further upstream bridges are located by the upstream bridge locating apparatus.

26. The apparatus of claim 25 wherein the bridge ID assigning mechanism comprises a mechanism that assigns a bridge ID to each located downstream bridge.

27. The apparatus of claim 26 wherein the information entering mechanism comprises a mechanism that enters information into internal registers and address bitmap of at least one downstream bridge when no further downstream bridges are detected by the downstream bridge locating mechanism.

28. The apparatus of claim 18 further comprising a mechanism that walks the bus system to discover upstream bridges and a mechanism that enters information into internal registers and address bitmap of each discovered upstream bridge to control the flow of information between bus segments.

29. The apparatus of claim 18 wherein the bridge ID assigning mechanism comprises a reset device that sets the bridge ID of all bridges to a predetermined

bridge ID upon reset and the configuration host comprises a mechanism that successively configures each bridge by sending commands and data to the predetermined bridge ID.

30. The apparatus of claim 18 wherein the bridge ID assigning mechanism comprises CFG IN/CFG OUT pins on each bridge wherein all bridges on the same hierarchical level have their CFG IN/CFG OUT pins connected together so that only one bridge at a time responds to the predetermined bridge ID.

31. The apparatus of claim 30 wherein the CFG IN/CFG OUT pins of all bridges on the same hierarchical level are connected in a daisy chain configuration.

32. The apparatus of claim 31 wherein a bridge in the daisy chain configuration enables the next bridge in the daisy chain configuration to respond to the predetermined bridge ID when the bridge is assigned a bridge ID other than the predetermined bridge ID.

33. The apparatus of claim 18 wherein at least some of the bridges are bi-directional bridges comprised of two unidirectional bridges connected in parallel and wherein the bridge ID assigning mechanism comprises a mechanism that assigns the two unidirectional bridges different bridge IDs.

34. The apparatus of claim 18 further comprising a mechanism that provides additional information to each bridge to enable the bridge to operate with a deterministic arbitration protocol.

35. A computer program product for configuring a bus system having a plurality of bus segments with bus master devices and slave devices connected thereto, the

bus segments connected by bus bridges, each bus bridge having a bridge ID, a plurality of internal registers and an address bitmap for controlling information flow through the bridge, the computer program product comprising a computer usable medium having computer readable program code thereon, including:

program code that walks the bus system to discover the bus topology and the bus bridges that form that topology;

program code that assigns a bridge ID to each discovered bridge; and

program code that enters information into internal registers and address bitmap of each discovered bridge to control the flow of information between bus segments.

36. A computer data signal embodied in a carrier wave for configuring a bus system having a plurality of bus segments with bus master devices and slave devices connected thereto, the bus segments connected by bus bridges, each bus bridge having a bridge ID, a plurality of internal registers and an address bitmap for controlling information flow through the bridge, the computer data signal comprising:

program code that walks the bus system to discover the bus topology and the bus bridges that form that topology;

program code that assigns a bridge ID to each discovered bridge; and

program code that enters information into internal registers and address bitmap of each discovered bridge to control the flow of information between bus segments.